Direct Transfer Functions

With the Direct Transfer Functions, you can unload and load source-code objects by using the Natural RPC (Remote Procedure Call) facility and the SYSRPC Utility as described in the relevant documentation.

When you invoke the Direct Transfer Functions, you can select the following:

- Direct Transfer using RPC
- Restart Direct Transfer
- Get Report from Direct Transfer Load
- Define Local TRANSFER System
- Preparing to Use Natural RPC for a Direct Transfer

Direct Transfer using RPC

A direct transfer starts a remote server, which unloads local data to a remote work file (Work File 1) and then loads the data into the corresponding remote environment.

Before you select the objects to be transferred on the Unload Objects menu, you have to specify the following direct transfer options for defining your transfer session with Natural RPC:

Direction	**CUIs only.** An output field that specifies the transfer direction (for future use).
Remote TRANSFER System Number	A four-digit code (*nnnn*) identifying the location of the remote subprogram TRPC*nnnn*, which is to perform the transfer service. **Note:** So that you can perform a remote transfer, a client stub of the same name as the subprogram TRPC*nnnn* must be available; see the section Preparing to Use Natural RPC for a Direct Transfer.
Unload Options	For the individual unload options, refer to the General Unload Options in the section Unload Function.
Load Options	For the individual load options, refer to the General Load/Scan Options in the section Load, Scan and Restart Load Functions. **Note:** As Software AG'S Entire Broker is involved when using Natural RPC, data conversion is automatically done by the Entire Broker and not by SYSTRANS. Therefore, selecting the option "Load with User Conversion Table" means that in addition to the Entire Broker data conversion, a table is used to replace certain characters by others. This table (SULAS-AS or SULEB-EB) is part of the program SULCONV located in your remote environment.
Replace	If you specify Y (GUIs: Yes) and an object with the same name as the one you are loading already exists in the remote target environment, the target object is replaced.

GUIs:

Once you have specified the direct transfer options, you can select one of the following object types to be unloaded for the direct transfer:

Natural Objects
Maps
DDMs
Adabas FDT
Error Message Texts
Command Processors
Library

After each unload operation, you are returned to the Direct Transfer Function dialog box, which now shows a Load button (instead of the Exit button) that you must choose to explicitly start the load operation. You can perform multiple unload operations before you start the load operation.

CUIs:

Once you have specified the direct transfer options, the Unload Objects menu (see Objects to be Unloaded in the section Unload Function) is displayed, where you can select the objects to be unloaded.

As the load operation starts automatically after the unload operation is finished, the Load function is not explicitly invoked and you therefore have to specify the individual load parameters before you start unloading your objects. Therefore, when you return to the Unload Object menu, Load (instead of Exit) is assigned to PF3, which means that pressing PF3 both ends the unload session and starts the load operation.

Restart Direct Transfer

You use the Restart Direct Transfer function to resume a direct transfer that has been terminated abnormally. When you select this function, you first have to specify your Remote TRANSFER System Number (see Direct Transfers using RPC).

To be able to restart a remote transfer:

- A client stub of the same name as the subprogram TRPC*nnnn* must be available; see Preparing to Use Natural RPC for a Direct Transfer.
- The direct transfer load option Save Restart Information (see General Load/Scan Options) must have been selected. Otherwise, the direct transfer cannot be resumed and a window (GUIs: dialog box) is displayed showing you a corresponding message and asking you whether to perform the entire load operation again. If so, the Direct Transfer Reload window (GUIs: dialog box) is displayed, in which you can again specify the relevant load options as well as the Replace option.

See the Restart Load function for further information on restarting a data transfer.

Get Report from Direct Transfer Load

You use the function "Get Report from Direct Transfer Load" to obtain the load report, which is written to Work File 4 in the remote environment.

If you select this function, a window (GUIs: dialog box) is displayed in which you enter your Remote TRANSFER System Number (see Direct Transfers (using RPC)). Then, press ENTER (GUIs: choose OK) to have the contents of Work File 4 displayed on a screen (GUIs: list box). It displays a listing of the objects that have been loaded.

Define Local TRANSFER System

This function is only available in environments which support the Natural RPC servers.

You use the Define Local TRANSFER System function to specify your own TRANSFER system number with which you can be addressed as a server by other clients. You can define several servers (that is, TRANSFER systems), but only one system number per server.

When you select this function, you have to specify a four-digit Local TRANSFER System Number (*mmmm*) along with the DBID (database identification) and FNR (file number) of the target system file. The subprogram TRPC*mmmm* is then copied into the library SYSTEM on the specified system file.

Preparing to Use Natural RPC for a Direct Transfer

To use Natural RPC to perform a direct transfer

1. Define your TRPC*nnnn* subprogram in the service directory.
2. Generate the corresponding client stub and copy the generated stub in the library SYSTRANS.
   The client stub must have the same name as the TRPC*nnnn* subprogram; the parameters to be passed are provided in the parameter data area TRPCPDA, which is delivered in source form.
   For further information see Creating Stub Subprograms (Natural RPC documentation) and the SYSRPC Utility documentation.
3. Start the corresponding server.

